

# 千寻差分数据 Linux SDK 开发指南

V1.3.0

千寻位置网络有限公司

2018 年 11 月·上海

# 法律声明

版权所有© 2016，千寻位置网络有限公司。保留一切法律权利。本文档包含的所有内容除特别声明之外，版权均属于千寻位置网络有限公司所有，受《中华人民共和国著作权法》及相关法律法规和中国加入的所有知识产权方面的国际条约的保护。未经本公司书面许可，任何单位和个人不得以任何方式(电子或机械,包括影印)或理由对该文档或其包含的任何产品、服务、信息、材料的任何部分进行使用、复制、修改、抄录、传播或与其它产品捆绑使用、销售，否则将视为侵权，本公司必依法追究其法律责任。本文档并不代表供应商或其代理的承诺，千寻位置网络有限公司可在不作任何申明的情况下对本文档内容进行修改。本文档中提到的其它公司及其产品的商标所有权属于该商标的所有者。

**千寻位置网络有限公司**

**联系邮箱：**[service@qxdz.com](mailto:service@qxdz.com)

**官方网站：**[www.qxdz.com](http://www.qxdz.com)

## 目 录

第一章 概述 .....	6
1.1 产品简介 .....	6
第二章 开发指南 .....	7
2.1 开发准备 .....	7
2.1.1 注册为开发者 .....	7
2.1.2 创建应用 .....	7
2.1.3 下载 SDK .....	7
2.2 接入流程图 .....	7
2.2.1 使用流程 .....	8
2.3 配置工程 .....	8
2.3.1 so 和头文件引入 .....	8
第三章 SDK 接口说明 .....	10
3.1 配置 RTCM .....	10
3.1.1 接口定义 .....	10
3.1.2 示例代码 .....	10
3.2 启动 RTCM .....	11
3.2.1 接口定义 .....	11
3.2.2 示例代码 .....	11
3.3 获取 RTCM 数据 .....	12
3.3.1 接口定义 .....	12

3.3.2 示例代码.....	12
3.4 发送 GGA.....	13
3.4.1 接口定义.....	13
3.4.2 示例代码.....	13
3.5 获取 SDK 返回状态.....	13
3.6 获取用户账号信息.....	13
3.6.1 接口定义.....	13
3.6.2 示例代码.....	14
3.7 返回状态码.....	15
3.8 基本数据结构.....	16
3.8.1 差分流数据.....	16
3.8.2 差分 SDK 配置.....	17
第四章 常见问题.....	18
4.1 账号问题.....	18
4.2 无法获取差分数据.....	18
4.1.1 无网络启动差分 SDK.....	18
4.1.2 中途长时间断网.....	18
4.1.3 账号互踢.....	18
4.1.4 差分帐号过期.....	19
4.1.5 发送 GGA 格式不合法.....	19
4.1.6 开始拿到差分数据，后来拿不到差分数据.....	19
4.1.7 差分区域不支持.....	19

<b>4.1.8 差分数据无法打印</b> .....	19
<b>4.3 拿到差分数据，但无法使用</b> .....	19
<b>4.4 验证是否成功使用差分数据</b> .....	20

# 第一章 概述

## 1.1 产品简介

千寻位置网差分数据 Linux SDK 为基于嵌入式 Linux 平台的应用提供自动注册差分账号、自动获取差分数据的服务接口包，专注于为广大开发者提供高精度位置服务，本文简称为**差分数据 SDK**。

该 SDK 获取差分数据后，差分数据传输、芯片写入工作完全交由开发者完成，为开发者提供最大灵活性。用户在获取这些服务的同时，不用考虑数据存储，数据安全和繁琐的网络交互等细节。

千寻位置网差分数据 SDK 主要提供以下功能：

用户使用自己的高精度芯片差分算法，使用 SDK 可以获取差分数据，用户可以使用这些差分数据灌进相关定位芯片或算法。

## 第二章 开发指南

### 2.1 开发准备

#### 2.1.1 注册为开发者

开发者需要入驻千寻位置网，在千寻位置官网（[qwxz.com](http://qwxz.com)）注册为千寻位置开发者，注册成功并通过个人/企业用户认证后，便可成为千寻位置网的开发者。详细信息请见千寻官网帮助文档。

（<https://www.qwxz.com/help-document-settled-platform.html#link-0>）

#### 2.1.2 创建应用

登录千寻位置网官网的开发者管理控制台创建应用，成功后会自动生成该应用的 AppKey 和 AppSecret，并确保有足够的差分账号配额；AppKey 和 AppSecret 是应用的凭证信息，请妥善保管。详细信息请见千寻官网帮助文档。

#### 2.1.3 下载 SDK

在千寻位置网官网下载千寻差分数据 SDK（Linux 版）。

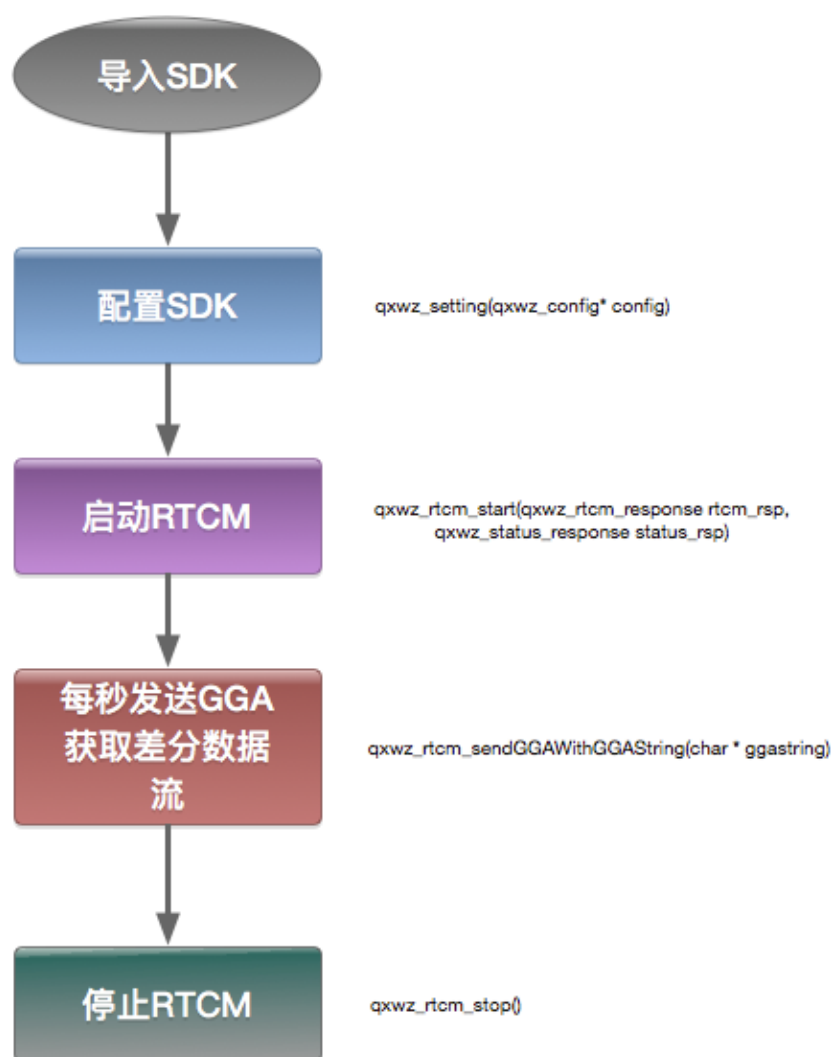
### 2.2 接入流程图

说明：目前 Linux 版差分 SDK 是通过 gcc 和 arm-none-linux-gnueabi-gcc 这两个编译器编译出来的。如有特殊平台需要，请联系服务邮箱 [service@wz-inc.com](mailto:service@wz-inc.com)

### 2.2.1 使用流程

1, 工程引入 SDK 包, 调用对应的 API 接口完成操作。

2, 由于 API 接口有先后依赖关系, 列出相关 API 的调用流程 (包括获取差分账号)。如下:



## 2.3 配置工程

### 2.3.1 so 和头文件引入

1, 准备千寻位置网提供的 so 包. 例如 "librtcm\_v0.0.1.so"



2, 将千寻位置网提供的 SDK 拷贝到项目目录下, 拷贝完成的工程目录如图所示:

```
-rwxrwxr-x  1 carry carry 149040 Jun 17 00:11 librtcm_arm_v1.0.0.so
-rwxrwxr-x  1 carry carry   9300 Jun 17 01:02 main
-rw-rw-r--  1 carry carry    937 Jun 17 00:12 main.c
-rw-r--r--  1 carry carry   1555 Jun 17 00:13 qxwz_rtcn.h
carry@ubuntu:~/c$ gcc -g -o main main.c -lrtcm -L. -Wl,-rpath=. -std=c99
```

## 第三章 SDK 接口说明

### 3.1 配置 RTCM

#### 3.1.1 接口定义

```
/**
 * 配置SDK
 */

void qxwz_setting(qxwz_config* config);
```

#### 3.1.2 示例代码

```
qxwz_config config;
/**
 * qxwz官网购买服务获取的appKey
 */
config.appkey=" ";
/**
 * qxwz官网购买服务获取的appSecret
 */
config.appSecret=" ";
/**
 * 设备的唯一ID，建议使用设备的序列号
 */
config.deviceId="00:11:67:44:41:6A ";
/**
 * 由调用者配置，一般取硬件对应的类型
 */
config.deviceType=" Test device";
qxwz_setting(&config);
```

## 3.2 启动 RTCM

### 3.2.1 接口定义

```
/**
 * 启动rtcm服务
 */

void qxwz_rtc_start(qxwz_rtc_response rtc_rsp, qxwz_status_response status_rsp);
```

### 3.2.2 示例代码

```
/*
 * 用户实现回调函数获取SDK差分数据，保证该回调函数最长阻塞时长不超过1秒。
 */
void qxwz_rtc_response_callback(qxwz_rtc data){
    printf("QXWZ_RTCM_DATA_LEN: %ld\n", data.length);
    printf("QXWZ_RTCM_DATA:");
    if (!data.buffer) {
        printf("no rtc data\n");
        return;
    }
    /*
     * 差分数据是二进制流，SDK集成者可通过data.length、data.buffer获取差分数据，将数据注入
     * 支持差分解算的定位模块即可出差分解。下面的示例代码仅以每行32字节格式打印差分数据。
     */
    int idx = 0;
    for (idx = 0; idx < data.length; idx++) {
        if (idx % 32 == 0)
            printf("\n");
        printf("%02x ", (unsigned char)(data.buffer[idx]));
    }
    printf("\n");
}

/*
 * 用户实现回调函数获取SDK状态码。
 */
void qxwz_status_response_callback(qxwz_rtc_status_code code){
    printf("QXWZ_RTCM_STATUS:%d\n",code);
}

qxwz_rtc_start(qxwz_rtc_response_callback,qxwz_status_response_callback);
```

## 3.3 获取 RTCM 数据

### 3.3.1 接口定义

```
/**
 * RTCM差分数据回调函数
 * @param data，差分数据结构体，data.length表示差分数据大小、data.buffer存储差分二进制数据流。
 */

typedef void (*qxwz_rtc_response)(qxwz_rtc data);
```

### 3.3.2 示例代码

```
/*
 * 用户实现回调函数获取SDK差分数据，保证该回调函数最长阻塞时长不超过1秒。
 */
void qxwz_rtc_response_callback(qxwz_rtc data){
    printf("QXWZ_RTCM_DATA_LEN: %ld\n", data.length);
    printf("QXWZ_RTCM_DATA:");
    if (!data.buffer) {
        printf("no rtc data\n");
        return;
    }
    /*
     * 差分数据是二进制流，SDK集成者可通过data.length、data.buffer获取差分数据，将数据注入
     * 支持差分解算的定位模块即可出差分解。下面的示例代码仅以每行32字节格式打印差分数据。
     */
    int idx = 0;
    for (idx = 0; idx < data.length; idx++) {
        if (idx % 32 == 0)
            printf("\n");
        printf("%02x ", (unsigned char)(data.buffer[idx]));
    }
    printf("\n");
}
```

## 3.4 发送 GGA

### 3.4.1 接口定义

```
/**
 * 向千寻位置的ntrip服务器发送GGA字符串用来获取rtcm数据
 * @param ggastring , GGA数据帧的结构如下:
 *   $GPGGA,<1>,<2>,<3>,<4>,<5>,<6>,<7>,<8>,<9>,M,<10>,M,<11>,<12>*xx<CR><LF>
 *   数据帧结束符<CR><LF>需保留。
 */

void qxwz_rtcm_sendGGAWithGGAString(char* ggastring);
```

### 3.4.2 示例代码

```
qxwz_rtcm_sendGGAWithGGAString("$GPGGA,000001,3112.518576,N,12127.901251,E,1,8,1,0,M,-32,
M,3,0*4B\r\n");
```

## 3.5 获取 SDK 返回状态

```
/**
 * RTCM服务状态码回调函数
 * @param code , SDK状态码 , 具体定义请参考头文件。
 */

typedef void (*qxwz_status_response)(qxwz_rtcm_status code);
```

## 3.6 获取用户账号信息

### 3.6.1 接口定义

```
/**
 *
 * 查询用户账号信息
 *
 * @return qxwz_account_info*
 */
```

```
*/  
qxwz_account_info* getqxwzAccount(void);
```

当用户账号还有 10 天将到期时，SDK 会通过状态码的方式通知用户账号即将到期，状态码为 2011。

当用户账号已过期时，SDK 会通知用户账号已过期，状态码为 2010。详见下一小节

### 3.6.2 示例代码

```
/**  
 * SDK 账号信息  
 */  
typedef struct {  
    char *appkey;  
    char *deviceID;  
    char *deviceType;  
    time_t expire_time; /*自1970年1月1日的秒数*/  
    char *NtripUserName;  
    char *NtripPassword;  
} qxwz_account_info;
```

```
qxwz_account_info *p_account_info = NULL;  
void get_qxwz_sdk_account_info(void) {  
    p_account_info = getqxwzAccount();  
    if(p_account_info->appkey != NULL) {  
        printf("appkey=%s\n",p_account_info->appkey);  
    }  
    if(p_account_info->deviceID != NULL) {  
        printf("deviceID=%s\n",p_account_info->deviceID);  
    }  
    if(p_account_info->deviceType != NULL) {  
        printf("deviceType=%s\n",p_account_info->deviceType);  
    }  
  
    if(p_account_info->NtripUserName != NULL) {  
        printf("NtripUserName=%s\n",p_account_info->NtripUserName);  
    }  
}
```

```
if(p_account_info->NtripPassword != NULL) {
    printf("NtripPassword=%s\n",p_account_info->NtripPassword);
}
printf("expire_time=%d\n",p_account_info->expire_time);
}
```

### 3.7 返回状态码

状态码	状态说明
1000	ntrip 连接到服务器
1001	ntrip 断开服务器
1002	appKey 验证失败
1003	appKey 验证成功
1004	网络异常
1005	NTRIP 用户达到上限
1006	NTRIP 用户不存在
1007	NTRIP 用户认证成功
1011	发送的 GGA 不合法
1012	GGA 发送超时
1013	正在连接 ntrip 服务器
1014	正在接收 ntrip 服务器数据
1015	非法 APP KEY
1016	非法 APP SECRET
1017	非法 Device type
1018	非法 Device id

1019	无法获取差分用户
1020	SDK 内部错误
1021	Ntrip 播发数据正常
1022	Ntrip 认证失败
1051	账号激活成功
1052	账号激活失败
1053	设备 UTC 时间设置不正确
2001	缺少参数
2002	账号不存在
2003	账号重复
2004	错误密码
2005	账号不可用
2006	没有有效的账号
2007	POPUser 不存在
2008	服务端内部错误
2010	账号已过期，需续费
2011	账号即将过期
2012	账号没有设置自动绑定

## 3.8 基本数据结构

### 3.8.1 差分流数据

/\*\*



```
* qxwz_rtcn 数据，包括数据，二进制格式 ( buffer )，和数据长度(length)
*/

typedef struct{
    char* buffer;
    size_t length;
}qxwz_rtcn;
```

### 3.8.2 差分 SDK 配置

```
/**
 * 启动SDK配置信息
 * appKey / appSecret  qxwz官网购买服务获取
 * deviceId 设备的唯一ID，建议使用设备的序列号
 * deviceType 设备类型
 */

typedef struct{
    char * appkey;
    char * appSecret;
    char * deviceId;
    char * deviceType;
}qxwz_rtcn_setting;
```

## 第四章 常见问题

### 4.1 账号问题

- (1) SDK 返回 1002、1015 表示 appKey 无效，请输入正确的 appKey/appSecret。
- (2) 差分 SDK 返回 2002 表示设备未绑定账号，请绑定当前使用的 deviceId 和 deviceType；
- (3) 差分 SDK 返回 2005 表示账号需要手动激活，请激活该 appKey 下的差分账号；
- (4) 差分 SDK 返回 2006 表示没有可用的账号，appKey 用户量已经达到上限，请在 appKey 账号里面申请扩容，允许更多用户接入差分服务；
- (5) 差分 SDK 返回 2012 表示设备服务号在该 sik 下只能手动绑定。
- (6) 差分 SDK 返回 2010 表示账号已过期。

### 4.2 无法获取差分数据

#### 4.1.1 无网络启动差分 SDK

差分 SDK 会返回 1004 (无网络状态)，请打开设备网络，确保连接网络，差分 SDK 需要网络。

#### 4.1.2 中途长时间断网

设备断网瞬间差分 SDK 返回 1004 (设备网络异常)、1001 (断开与 Ntrip 服务器的连接)，差分 SDK 会尝试再次连接服务器，如果成功连接服务器状态返回 1000，如果长时间处于重连状态，SDK 始终上报 1013 (正在连接 Ntrip 服务器)，请检查网络情况。

#### 4.1.3 账号互踢

同时刻使用同一个账号启动差分 SDK，SDK 会上报状态码 2009 (Ntrip 服务器断开 Socket 连接)，表示同一个账号出现了竞争互踢现象，集成开发者需保证同一账号不在同时刻启动使用。

#### 4.1.4 差分帐号过期

差分 SDK 会返回 2010 (账号已过期需续费)，需开发者登陆千寻位置官网进行续费操作。

#### 4.1.5 发送 GGA 格式不合法

正常启动差分 SDK 后，差分 SDK 返回 1000 (连接服务器成功)，调用 sendGGA 方法后，拿不到差分数据，请检查发送的 GGA 字符串格式是否正确，数据帧结束符<CR><LF>需保留。

#### 4.1.6 开始拿到差分数据，后来拿不到差分数据

正常启动差分 SDK 后，开始能正常拿到差分数据，过段时间后拿不到差分数据，这是因为差分 SDK 必须在 120 秒内向服务器传递最新位置，如果在 120 秒内没有给服务器最新位置，服务器将不会播发差分数据。为了正常拿到差分数据，建议您每秒向差分 SDK 发送最新位置。

#### 4.1.7 差分区域不支持

正常启动差分 SDK 后，差分 SDK 返回 1000 (连接服务器成功)，调用 sendGGA 方法后，拿不到差分数据，是因为你发送的经纬度所在地区暂时没有提供差分服务。您可在千寻官网查询服务覆盖范围：<https://www.qxwz.com/map-cover.html>。如果您有疑问请联系千寻客服。

#### 4.1.8 差分数据无法打印

差分数据回调函数上报数据为二进制格式，无法以字符串方式打印，如需查看内容，请以十六进制格式打印。

### 4.3 拿到差分数据，但无法使用

(1) 请先确认您使用的设备是 RTD 设备或者 RTK 设备，RTD 设备只能支持 FindM 服务，RTK 设备只能支持 FindCM 服务。

(2)请确认您的设备支持的差分数据类型是 RTCM32 还是 RTCM23。若您的设备支持 RTCM32 数据结算, 还请使用 RTCM32 差分 SDK, 若您的设备支持 RTCM23 数据解算, 还请使用 RTCM23 差分 SDK。

## 4.4 验证是否成功使用差分数据

根据 NMEA 语句判断当前是否为高精度解算位置。开发者可根据定位模块吐出的 NMEA 语句进行判断。如下:

\$GPGGA,<1>,<2>,<3>,<4>,<5>,<6>,<7>,<8>,<9>,M,<10>,M,<11>,<12>\*xx, 当第<6>个字段表示 GPS 状态, 出现以下三类之一时: 2 (码差分), 4 (固定解), 5 (浮点解), 代表当前是高精度解算位置。如果长时间未进入高精度定位, 请确认设备在开阔无遮挡空间测试, 确定 GNSS 信号强度足够。